Attorney Docket No. 1497/2

UNITED STATES PATENT APPLICATION

METHODS AND SYSTEMS FOR PER-SESSION DYNAMIC MANAGEMENT OF MEDIA GATEWAY RESOURCES

Inventors: Edward Y. Qian, Plano, Texas
David Z. Lu, Dallas, Texas
San-Qi Li, Plano, Texas
Yen Shei, Plano, Texas
Weijun Lee, Plano, Texas

Assignee: Santera Systems, Inc.

Entity: Large Entity

JENKINS, WILSON & TAYLOR, P.A.
Suite 1400, University Tower
3100 Tower Boulevard
Durham, North Carolina 27707
Telephone: 919-493-8000
Facsimile: 919-419-0383

"Express Mail" mailing number ER467276785US
Date of Deposit October 1, 2003
I hereby certify that this correspondence is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" service under 37 C.F.R. 1.10 on the date indicated above and is addressed to Mail Stop Patent Application, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

Shaylor E. Dunn

Description

METHODS AND SYSTEMS FOR PER-SESSION DYNAMIC MANAGEMENT OF MEDIA GATEWAY RESOURCES

Technical Field

The present invention relates to methods and systems for media gateway (MG) resource allocation. More particularly, the present invention relates to methods and systems for dynamically allocating media gateway resources from a shared pool of media gateway resources on a per call (i.e., per session) basis.

Related Art

In modern telephony networks, media switching and call control functionality are separated. Call control, which includes setting up and tearing down calls and maintaining call state machines, is performed by a network entity referred to as a media gateway controller (MGC). Media stream switching, which includes switching media packets between input and output ports and converting the media packets into the appropriate formats for the sending and receiving parties, is performed by a media gateway (MG). Media gateway controllers communicate call control information to media gateways via a media gateway control protocol. Typical media gateway control protocols,

such as MGCP and MEGACO, include commands for communicating information about each endpoint of a session to the media gateway and instructing the media gateway as to how to process packets to be delivered to each endpoint.

Conventional media gateways statically bind logical and physical voice-processing resources (e.g., IP addresses, UDP ports, CODECs, SAR chips, DSPs, etc.) to network interface cards (NICs) connected to external networks either at the data link layer (OSI Layer 2) or the IP network layer (OSI Layer 3). Because of the static nature of conventional methods for allocation of media gateway resources, external network topology changes require static reallocation of voice channels within media gateways. Such static resource reallocation is labor-intensive and unsuitable for dynamically changing network environments. For example, a voice chip resource in a media gateway is conventionally assigned a block of static IP addresses through manual provisioning, and each of these static IP addresses is reserved for a remote network. Voice chip resources are not pooled together to serve any one of the remote networks as needed. As a result, the voice-processing capacity available for each remote network is constrained by the data link layer bindings to individual voice processing chips.

Figure 1 is a block diagram illustrating conventional resource allocation in a media gateway. Referring to Figure 1, media gateway **100** includes a plurality of voice server cards **102**. Each voice server card **102** includes a plurality of voice chips **104**. Each voice chip **104** is assigned to serve one of

the external networks **106**, **108**, **110**, **112**, **114**, or **116**. A media gateway controller **118** controls media gateway **100** to establish and tear down calls with entities in external networks **106-116**.

In the exemplary configuration illustrated in Figure 1, each external network is statically assigned a single voice chip capable of simultaneously processing K voice channels. The one-to-one assignment is shown in Figure 1 for illustration only. In actual network implementations, each external network may be statically assigned multiple voice chips, and the processing capability of each voice chip may be statically split and reserved between multiple remote networks. Nevertheless, all these mappings are statically assigned at the IP address level, i.e. at the network layer.

In the example illustrated in Figure 1, media gateway **100** may include a static resource allocation table that allocates voice chips **104** to external networks **106-116**. During signaling required to set up a call, media gateway controller **118** communicates information to external networks **106-116** regarding the IP addresses of the voice chip and network interface card to which the voice packets should be addressed based on the static resource allocation table.

One problem with this allocation scheme is that if one of the external networks wishes to reconfigure or add additional endpoints, the provisioned assignments at media gateway **100** and/or MGC **118** must be reconfigured. For example, if the owners of external network **108** need to add a new VLAN, the resource allocation table in MG **100** would have to be reconfigured to

support the new VLAN. Such manual re-provisioning is labor-intensive and unsuitable for dynamically changing networks.

Another problem with the conventional media gateway resource allocation scheme illustrated in Figure 1 is that voice resources are not efficiently utilized. For instance, since voice chips are statically assigned to serve external networks, load sharing of voice chip resources among different networks cannot be performed. As a result, the load on the voice chips can be unbalanced and calls from one network can be blocked even though voice chip resources assigned to serve another network are available.

Accordingly, in light of the difficulties associated with conventional static allocation of media gateway resources at OSI Layer 2 or OSI Layer 3, there exists a need for improved methods and systems for allocating resources at a media gateway.

Disclosure of the Invention

The present invention includes improved methods and systems for per-session dynamic management of media gateway resources. The present invention offers a unique method to dynamically and efficiently allocate and interconnect a media gateway's logical and physical resources for each Voice-over-IP (VoIP) call at the UDP port level, i.e. at OSI Layer 4.

The logical resources for a VoIP call include two tuples: the <local IP address for RTP, local UDP port for RTP> pair and, optionally, the <local IP address for RTCP, local UDP port for RTCP> pair. The physical resources for a VoIP call include Network Interface Cards (NICs), a VoIP chip channel, and,

optionally, an echo cancellation channel, a transcoding channel, a tone/announcement channel, etc.

Unlike conventional static resource management methods, which bind each local IP address to a specific VoIP chip (in general, a media processing unit) and a specific NIC on the media gateway, the present invention disassociates each local IP address from any specific VoIP chip or NIC. Instead, the present invention allocates and manages the physical and logical resources for a VoIP call by the logical UDP ports within each IP address, i.e. by the <local IP, local UDP> pair, which results in a finer granularity than by the IP address alone. A <local IP, local UDP> pair may be served by any available VoIP chip and may be reachable from multiple NICs via pre-provisioned internal data paths. In one exemplary configuration, each VoIP chip may be reachable from all of the NICs in a media gateway.

To set up a VoIP call, a <local IP, local UDP> pair is allocated from the logical resource pool and assigned to one serving VoIP chip channel. The <local IP, local UDP> pair is communicated to some or all NICs, depending on configured network topology. When an IP packet is received by any of the NICs, the NIC looks up the packet's <local IP, local UDP> pair to identify its serving VoIP chip and the internal data path to that chip. The VoIP chip maps each <local IP, local UDP> pair to its assigned VoIP chip channel.

Rather than statically allocating voice server resources to external networks, a media gateway pools voice server resources provided by a plurality of voice chips in the media gateway and dynamically allocates resources from the pool on a per call or per-session basis. For example, when a new session

is requested from the media gateway controller, the media gateway assigns a voice chip to the session independently of the network from which the session originates. Media packets associated with the session are processed by the assigned voice chip. New sessions from the same network may be assigned to the same voice chip or to different voice chips, depending on the load-sharing algorithm used by the dynamic resource manager of the media gateway. Because media gateway resources are dynamically allocated from a common pool, reprovisioning is not required when the topology of one or more external networks changes. In addition, media gateway resources can be more efficiently utilized.

Accordingly, it is an object of the invention to provide methods and systems for dynamic media gateway resource management.

It is another object of the invention to provide a method for dynamically allocating media gateway resources on a per-call (i.e. per-session) basis.

Some of the objects of the invention having been stated hereinabove, and which are addressed in whole or in part by the present invention, other objects will become evident as the description proceeds when taken in connection with the accompanying drawings as best described hereinbelow.

Brief Description of the Drawings

Preferred embodiments of the invention will now be explained with reference to the accompanying drawings of which:

Figure 1 is a block diagram illustrating conventional static management of media gateway resources;

Figure 2 is a block diagram illustrating per-session dynamic management of media gateway resources according to an embodiment of the present invention;

Figure 3 is a block diagram illustrating an exemplary internal architecture of a media gateway in which the methods and systems of the present invention may be implemented;

Figure 4 is a flow chart illustrating a method for dynamic management of media gateway resources according to an embodiment of the present invention;

Figure 5 is a block diagram illustrating exemplary rerouting of a voice stream when a network interface fails according to an embodiment of the present invention; and

Figure 6 is a block diagram illustrating exemplary incoming and outgoing packet processing by a media gateway according to an embodiment of the present invention.

Detailed Description of the Invention

Figure 2 illustrates an example of a media gateway including a dynamic resource manager according to an embodiment of the present invention. Referring to Figure 2, media gateway **200** includes a dynamic resource manager **204** for dynamically allocating voice server resources. Dynamic resource manager **204** may be implemented in hardware, software, firmware,

or any combination thereof as part of a control module **202** within media gateway **200**.

In Figure 2, a packet matrix module **206** provides provisioned internal data-paths between the network interface cards **208** and every voice chip **104**. For clarity, the full mesh between voice chips **104** and NICs **208** is not shown. It is understood that packet matrix module **206** may provide paths such that each voice chip **104** is reachable from each NIC **208** and vice-versa.

Each of the data paths between voice chips **104** and NICs **208** may be individually provisioned based on total bandwidth, scheduling algorithm, and traffic management policy, etc. As a result, voice chips **104** can be considered a resource pool that can be assigned dynamically to any new session via any of the NICs **208**. In one implementation, the incoming connections may be terminated at NICs **208** at the data link layer (e.g. Ethernet, VLAN, ATM, MPLS) or the IP layer, and dynamic resource manager **204** in control module **202** dynamically assigns a voice chip from the pool of voice chips to process each incoming session/call.

Rather than statically assigning voice chips **104** to each external network, dynamic resource manager **204** dynamically allocates voice chips on a per-call (i.e. per-session) basis. As a result, each network is able to use the media-gateway-wide pool of M x N x K channels, where M is the number of voice over cards, N is the number of voice chips on each voice sever card, and K is the number of channels supported by each voice chip. The number of channels per voice chip, K, may be different for different codec types, e.g.

G.711, G.726, G.729, etc. In comparison, in the conventional static allocation schemes described above, each external network is only able to access only K or some statically configured number of channels in the media gateway and cannot access voice channels assigned to other external networks. Because every external network now has a shared pool of M x N x K dynamically assignable channels, reprovisioning of media gateway resources is not required when one of the networks adds additional endpoints. In addition, calls can be more evenly distributed among voice chips.

Figure 3 is a block diagram illustrating an exemplary internal architecture for media gateway **200** in more detail. In Figure 3, media gateway **200** includes voice servers **102**, as described above. Each voice server **102** has various voice chips, including voice-over-IP chips **302**, voice-over-AAL1 chips **304**, and voice-over-AAL2 chips **306**. In addition, each voice server includes some digital signal processors **308** (e.g. voice transcoders, echo cancellers, conference bridges, etc.), a time slot interconnection (TSI) **310**, a central processing unit (CPU) **312**.

In the illustrated example, each voice chip **302** implements one or more voice-over-IP protocols, such as Real time Transmission Protocol (RTP). Each voice chip **304** implements ATM Adaptation Layer 1 (AAL1) functions. Each voice chip **306** implements ATM Adaptation Layer 2 (AAL2) functions. DSP **308** provides transcoding, echo cancellation and other payload-transformation functions. TSI **310** makes on-demand connections between VoIP chip

channels, TDM matrix channels, and DSPs. CPU **312** controls the overall operation of each voice server module **102**.

In addition to voice server modules **102**, media gateway **200** includes a plurality of network interface cards **314**. Each network interface card **314** implements network layer functions and packet forwarding functions, such as IP forwarding functions. In the illustrated example, different network interface cards are provided to connect to external Ethernet, Packet-Over-SONET (POS), ATM, and MPLS networks.

In addition to packet-based network interface cards, media gateway **200** may also include TDM network interface cards **318**. TDM network interface cards **318** send and receive voice frames from external TDM networks. TDM network interface cards **318** may implement any suitable physical layer protocols for sending and receiving voice frames over TDM links. For example, each TDM NIC **318** may terminate one or more TDM voice trunks.

In order to switch media packets between network interface cards **314** and voice server modules **102**, media gateway **200** includes a packet matrix module **206**. Packet matrix module **206** switches packets under the control of control module **202**. As discussed above, packet matrix module **206** may connect every packet NIC to every voice chip **302**. In addition to packet matrix module **206**, gateway **200** may also include a TDM matrix module **322** for switching traffic that are carried in each TDM timeslot. TDM matrix module **322** is also controlled by control module **320**.

In addition to controlling the packet and TDM matrices, control module **202** implements dynamic voice server resource allocation according to the present invention. For example, control module **202** may communicate with an internal or external media gateway controller to dynamically allocate logical and physical resources for each call.

Figure 4 is a flow chart illustrating exemplary steps that may be performed by control module **202** in dynamically allocating voice server resources according to an embodiment of the present invention. Referring to Figure 4, in step **400**, control module **202** receives a request for a new call/session. The request may be generated by media gateway controller **118** in response to a call setup message associated with a new call. The call setup message may be an ISUP IAM message, a PRI SETUP message, a SIP INVITE message, or any other suitable type of call setup message for initiating a call. In step **402**, control module **202** checks available voice server resources from the shared pool of voice server resources to determine whether any resources are available. In step **404**, if resources are not available, control proceeds to step **406** where the call attempt is rejected.

If, in step **404**, control module **202** determines that resources are available, control proceeds to step **406** where control module **202** dynamically assigns voice server resources from a shared pool. Dynamically assigning voice server resources may include assigning a voice chip to process the media stream for the session. The voice chip is preferably selected independently of the remote network. Because there is no fixed association

between the remote network and the voice chip, each call/session has access to all of the available resources of media gateway **200**. As a result, voice server resources of media gateway **200** are more efficiently allocated than in conventional media gateways. In addition, when the topology of an external network changes, because resources at media gateway **200** are dynamically allocated, there is no need to manually reconfigure media gateway **200**.

In step **410**, control module **202** dynamically assigns a local IP address and a local UDP port to the RTP flow of the session. The combination of local IP address and UDP port number is unique among current sessions. The local IP address and UDP port combination is preferably assigned to a voice chip for the duration of the session. The local IP address and UDP port combination is communicated to the remote end of a session by media gateway controller **118**. The remote end of the session will then send subsequent media stream packets to the local IP address and UDP port combination. Packet forwarding tables on each packet network interface **314** are updated so that packets addressed to the local IP and UDP combination assigned to the voice server chip are forwarded to the appropriate voice chip. Because the forwarding tables at each NIC **314** are dynamically constructed and updated as calls are established and released, media gateway **200** is capable of adapting to dynamically changing network conditions.

Figure 5 illustrates dynamic resource allocation and <IP, UDP> address assignment according to the invention in more detail. In Figure 5, it is assumed that VPN1 **106** requests a first voice channel with gateway **200**. Gateway **200**

dynamically assigns a channel on voice chip 1 **104** to the session and associates the local IP address, UDP port pair <IP_1, UDP_1> to the voice chip. In one exemplary implementation, control module **202** multicasts the address combination < IP_1, UDP_1> to all NICs **314**. Each IP NIC **314** includes a packet-forwarding table **500** that is updated to reflect that the combination < IP_1, UDP_1> is associated with voice chip 1. Each IP NIC **314** includes its own IP address that it advertises to external nodes, such as external router **502**. The present invention places no specific requirements on any external router. An external router **502** may use any routing method to deliver IP packets to the media gateway. In the illustrated example, since both NICs **314** have routes to IP address IP_1, the external router **502** may use its own routing policies to select any route at any point of time.

If the path between external router **502** and IP NIC #1 **314** fails, the forwarding table in external router **502** will be updated to set the cost associated with the route via IP address IP_A to a predetermined large value. In this situation, external router **502** would reroute the voice stream packets to NIC #2 **314** directly or via other transmit routers. Because NIC #2 **314** includes a route to <IP_1, UDP_1>, interruption in service will not occur when the path to one of the IP NICs fails. In addition, because voice server resource allocation information is multicast among NICs **314**, failure of any NIC will not make the voice server resources unreachable from outside. In this scenario, IP routing protocols would notify external routers to update their forwarding tables to re-route media packets over one of the operational NICs. In static resource

allocation schemes, where each session uses the statically assigned IP address bound to a specific media gateway NIC, failure of the NIC would result in failure of the session. Thus, the present invention is more robust than conventional static media gateway resource allocation schemes.

Another advantage of the dynamic resource allocation schemes of the present invention is real-time adaptability to network topology changes. Unlike conventional media gateway implementations, if VPN1 **106** needs to establish a second connection with the media gateway, VPN1 **106** is not limited to utilizing voice chip 1 **104**. In the illustrated example, it is assumed that the second connection from VPN1 **106** is assigned to voice chip 4 **104**. The <IP, UDP> address combination assigned to the second session is assumed to be <IP_1, UDP_2>. In conventional media gateway implementations, subsequent sessions from VPN1 **106** would be routed to the same voice chip. By dynamically allocating voice chips and local <IP, UDP> address combinations, the present invention breaks the fixed association between voice server resources and external networks, thus allow newly added external terminals to access the existing voice server resources within a media gateway without any manual reconfiguration. As a result, gateway **200** illustrated in Figure 5 is more easily adaptable to subscriber changes in external networks than conventional media gateway implementations.

Returning to Figure 4, once a local IP address and UDP port combination has been assigned to a session and resources have been allocated for the session, media gateway **200** processes a session using the

associated resources. Processing the session using the associated resources first includes forwarding packets by their destination IP address and destination UDP port to the voice chip dynamically assigned to the session. The voice chip dynamically assigned to the session then performs voice processing on media packets associated with the session. Exemplary operations that may be performed by the assigned voice chip may include segmentation and reassembly (SAR), echo cancellation, transcoding, DTMF detection, DTMF generation, announcement, conference bridging, Internet Fax, and law enforcement. Once the voice packets associated with the session have been processed, the voice packets may be sent from the voice chip to one of the network interface cards **314** or to a TDM network interface card for transmission to the remote end of a session. If the remote end of a session is an IP connection, IP NICs **314** may route the outbound packet to the remote end of the session using IP forwarding tables **500**. Once a session ends, the local <IP, UDP> address combination assigned to the session may be returned to a common pool for assignment to a new session.

The present invention is not limited to identifying sessions using destination IP address and destination UDP port. In an alternate implementation of the invention, each session may be identified by a combination of destination IP address, destination UDP port, source IP address, and source UDP port. Using all four parameters to identify a session may be advantageous for firewall filtering purposes. Using any combination of

identifiers in session packets to uniquely identify sessions is intended to be within the scope of the invention.

Figure 6 illustrates the forwarding of inbound and outbound media packets in more detail. In Figure 6, an incoming packet **600** addressed to <IP_2, UDP_1> arrives at packet interface **314**. In this example, it is assumed that voice server resources have already been assigned to the session with which packet **600** is associated, and the packet interface **314** session has been notified by the control module of the established association. Because packet interface **314** is session-aware, packet interface **314** forwards the packet based on the <destination IP address, destination UDP port> combination to a voice chip associated with this combination. In the illustrated example, packet **600** travels to voice chip **104** over a pre-configured internal data path Path55 through the packet matrix to voice chip **104**. Voice chip **104** processes the packets associated with the session, switches the source and destination IP and UDP ports in the packets, and forwards outbound packet **602** to packet interface **314** over another pre-configured internal data path Path66. Packet interface **314** forwards outbound packet **602** to the next hop IP address associated with the remote end of the session. Thus, by making interfaces session-aware, the present invention provides increased packet interface functionality over conventional media gateway solutions in which packet interfaces only perform data-link layer (Layer 2) or IP layer (Layer 3) forwarding.

Thus, as described above, the present invention includes methods and systems for dynamic allocation of media gateway resources on a per session basis. Incoming packets are dynamically assigned to voice chip resources on a per session basis. Once a voice chip resource has been assigned to a session, resource allocation tables associated with packet interfaces are updated to include the session information. Subsequent packets associated with the session are then forwarded to the selected voice chip resource.

By breaking the static bindings between voice chip resources and external networks, the present invention reduces the impact of network topology changes on media gateway functionality. In addition, each new session may be served by any of the pooled M x N x K channels, where M is the number of voice over cards, N is the number of voice chips on each voice sever card, and K is the number of channels supported by each voice chip. The value of K depends on the actual codec type. Such pooled resources can be contrasted with conventional statically allocated media gateway implementations where sessions are statically assigned to voice chips and therefore only have K or some statically configured number of channels available. Finally, a media gateway according to the present invention has enhanced fail-over capabilities because resource assignment information is multicast to multiple packet interfaces. Thus, if a packet interface or a route to a packet interface fails, the voice chip resources associated with each session are still reachable through other packet interfaces.

It will be understood that various details of the invention may be changed without departing from the scope of the invention. Furthermore, the

foregoing description is for the purpose of illustration only, and not for the purpose of limitation, as the invention is defined by the claims as set forth hereinafter.